A COMPARISON OF DATA-DERIVED AND KNOWLEDGE-BASED MODELING OF PRONUNCIATION VARIATION

Mirjam Wester[1, 2] *& Eric Fosler-Lussier*[1]

[1]International Computer Science Institute, 1947 Center Street, Suite 600, Berkeley, CA 94704, USA
[2]*A*^{2}*RT*, Dept. of Language and Speech, University of Nijmegen, the Netherlands

ABSTRACT

This paper focuses on modeling pronunciation variation in two different ways: data-derived and knowledge-based. The knowledge-based approach consists of using phonological rules to generate variants. The data-derived approach consists of performing phone recognition, followed by various pruning and smoothing methods to alleviate some of the errors in the phone recognition. Using phonological rules led to a small improvement in WER; whereas, using a data-derived approach in which the phone recognition was smoothed using simple decision trees (d-trees) prior to lexicon generation led to a significant improvement compared to the baseline. Furthermore, we found that 10% of variants generated by the phonological rules were also found using phone recognition, and this increased to 23% when the phone recognition output was smoothed by using d-trees. In addition, we propose a metric to measure confusability in the lexicon and we found that employing this confusion metric to prune variants results in roughly the same improvement as using the d-tree method.

1. INTRODUCTION

Approaches to modeling pronunciation variation can be roughly divided into pronunciation variants being either derived from a corpus of pronunciation data or from pre-specified phonological rules based on linguistic knowledge [1]. In this study, we investigate both approaches. In addition to comparing the different WER results, we also compared the lexica obtained through the different approaches; to analyze how much of the same pronunciation variation is modeled by the approaches.

One of the problems that you encounter when modeling pronunciation variation, which holds for both the knowledge-based approach as well as the data-driven approach, is that the confusability within the lexicon increases when variants are added. This problem has been signaled by many researchers in the field of pronunciation variation [1]. Confusability is often introduced by statistical noise in phonetic transcriptions. One commonly used procedure to alleviate this is to smooth the phonetic transcriptions – whether provided by linguists [2] or phone recognition [3] – by using decision trees to limit the observed pronunciation variation. Other approaches [4, 5] combat confusability by rejecting variants that are highly confusable on the basis of phoneme confusability matrices, or for instance in [6] a maximum likelihood criterion is used to decide which variants to include in the lexicon.

However, in none of these approaches a measure for confusability is given. In this paper, we propose a metric that calculates the confusability in a lexicon given a set of training data. In first instance, the metric was intended only to compare confusability in lexica obtained through the different approaches. However, we also carried out experiments to see if the metric could be employed to reduce WERs.

2. SPEECH MATERIAL

In this study, we focus on segmental (phonetic) variation within VIOS [7], a Dutch database, which consists of recordings of interactions between man and machine in the domain of train timetable information. Our training and test material, selected from the VIOS database, consisted of 25,104 utterances (81,090 words) and 6,267 utterances (20,489 words), respectively.

3. LEXICA GENERATION

The starting point is the baseline lexicon (**1_Baseline**). It contains one pronunciation for each word. This lexicon is based on the baseline lexicon used at *A*^{2}*RT* [8]. All of the lexica described in the following sections were created by merging the baseline lexicon with the new variants. Prior probabilities for the variants were based on their frequency counts in the training data.

3.1. Knowledge-based lexicon

In a knowledge-based approach, the information about pronunciations is derived from knowledge sources, for instance handcrafted dictionaries or the linguistic literature. In this study, we selected five Dutch phonological processes, which are described in the literature, to formulate rules with which pronunciation variants were generated. The rules are context dependent and are applied to the words in the canonical lexicon. The resulting variants are added to the lexicon (**2_PhonRules**). Table 1 shows the five phonological rules and their contexts for application. For a detailed description of the processes see [8].

Rule	Context for application
/n/-deletion[1]	n → ∅ / @ ___ #
/r/-deletion	r → ∅ / [+vowel] ___ [+consonant]
/t/-deletion	t → ∅ / [+obstruent] ___ [+consonant]
schwa-deletion	@ → ∅ / [+obstruent] ___ [+liquid] [@]
schwa-insertion	∅ → @ / [+liquid] ___ [-coronal]

Table 1: Phonological rules + context for application.

[1] Sampa phoneme notation, see: http://www.phon.ucl.ac.uk/home/sampa/dutch.htm

3.2. Data-derived lexica

In a data-derived approach, the information used to develop the lexicon is in some way distilled from the data. The approach we use is similar to other methods used in the field: phone recognition is carried out on the training data to supply the raw information on pronunciations. In this type of recognition task, the lexicon does not contain words, but a list of 39 phones and a phone bigram grammar is used to provide phonotactic constraints. The output is a sequence of phones; no word boundaries are included. Therefore, the next task is to insert these boundaries. This is done by aligning the phone recognition output to a reference transcription that contains word boundaries. A distance measure based on binary phonetic features was employed to align the strings of phones and insert the word boundaries at the most appropriate places in the string.

These alignments are used as the base information for generating the data-derived lexica. First of all, we made a lexicon in which all the variants generated by the phone recognition were added to the baseline lexicon (**3_PhoneRec**).

One of the drawbacks of using the phone recognition output to generate new lexica in this way is that the phone transcriptions contain errors, which means a lot of "incorrect" transcriptions are included in the lexicon. To get an indication of how much of the output might be noise, we compared the phone recognition to the reference transcription, and found 68% phone accuracy. Although a great deal of this may be noise, we also know that a lot of reduction takes place in spontaneous speech so that part of the “errors” must be the pronunciation variation that we are interested in. Therefore, we sought for ways to eliminate the “incorrect” transcriptions whilst keeping the relevant information about pronunciation variation.

One of the techniques we used was to make a pre-selection of the utterances prior to generating the lexicon, instead of using all of the phone recognition output. The pre-selection criteria were based on the alignment between the phone recognition and the reference transcription. For each utterance the phone error rate was calculated. Using this information it was possible to incorporate the following two selection criteria:

- an utterance must contain less than 40% errors, and
- words with more than two deleted phones in a row were excluded.

Thus, a lexicon was created based on what we expect to be less noisy data. (**4_PhonRec_Sel**)

The other approach we used to remove some of the noise in the transcriptions was by using decision trees [10] to smooth the phone recognition before generating a lexicon. We used very simple decision trees (d-trees) in order to match the type of contexts used in our phonological rules. Thus, we did not use more complex features like syllable structure (as was done in [3]) but simply used the identity of the left and right phones as features.

In short, the method works as follows. For each of the 39 phones a d-tree was built. The d-tree model is trying to predict:

P (realization | left context, right context).

We allowed for automatic sub setting of feature values while generating the d-trees. Next, using the distributions in the d-trees, finite state grammars (FSG) were built for the utterances in the training data. Those FSG were realigned with the training data, and the smoothed phone transcriptions were used to generate a new lexicon. In order to compare the d-tree approach to the phone recognition approach we used the same selection criteria to restrict the data that was used as input to the d-trees. The resulting lexicon is referred to as **5_Dtree_Sel**.

3.3. A measure of confusability

As we mentioned in the introduction, one problem that we were concerned about was the addition of pronunciation variants that might make a word confusable with other words within the recognizer. We therefore created a metric by which we could judge the confusability of individual pronunciations, as well as the overall confusability of a lexicon.

The metric is calculated as follows: first a forced alignment of the training data is carried out using the lexicon for which the confusability is to be determined. Then, we compute the set of word pronunciations that match any sub string in the alignment, producing a lattice of possible matching words; this gives an overestimate of the confusability of the lexicon.

For example, in Figure 1, we compute the forced alignment of the word sequence "this is a test". We can then find all pronunciations in the lexicon that span any sub strings, e.g., the word "the" corresponding to the pronunciation "dh ih". The confusability metric is calculated by considering the number of words that correspond to each phone (as shown in Figure 1 in the row marked “All confusions”). The average confusability for the lexicon is then obtained by summing up the number of “confused” phones per phone and dividing by the total number of phones in the forced alignment. This is the average confusability that we present in the following section for the various lexica.

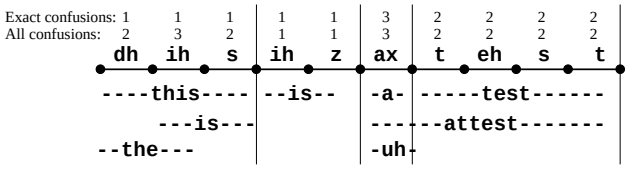

Figure 1: Example of part of the lattice used to compute the average confusion.

As described above, this metric overestimates the number of possible confusions, since it doesn't take into account that some words would be pruned during decoding because of a dead-end path in the word lattice: for example, the word "the" in Figure 1 doesn't have any appropriate following word in the lattice. The "exact confusion" metric ameliorates this somewhat by only counting confusions that occur at the word boundaries provided by the forced alignment. Since this is an underestimate of the amount of confusion in the lexicon, one can use this as a lower bound.

To investigate what the effect is of removing highly confusable variants we created two new lexica. First we took the lexicon 4_PhRec_Sel, and removed all words which had a confusion

count of over 100 (**6_ PhRec_Sel_100**). We did the same for 3_PhRec, resulting in **7_ PhRec_100**. We ensured that baseline variants were not removed from the lexica, in order to keep the comparison with the other lexica fair. Due to time constraints we have not yet carried out the same experiments for the knowledge-based approach.

4. CSR

All of the experiments were carried out with the ICSI hybrid ANN/HMM speech recognition system [10]. The baseline neural network was bootstrapped using alignments of the training material obtained with the baseline recognition system used at *A²RT* [8, 9]. The main difference between these two systems is that in the ICSI system acoustic probabilities are estimated by a neural network instead of by mixtures of Gaussians, as is the case in the *A²RT* system.

For the front-end acoustic processing we use 12th-order PLP features [11] and energy, which are calculated every 10 ms, for 25ms frames. The neural net uses the input features and additional context from eight surrounding frames of features to estimate the probability that the input corresponds to each of the defined categories. The categories that we use are 39 context-independent phones for Dutch. The neural network had a hidden layer size of 1000 units and the same network was employed in all experiments. Finally, a bigram language model was used which was also based on the *A²RT* alignments.

5. RESULTS

5.1. Lexica

In Table 2, the statistics for the various lexica are shown. The second column shows the number of words in the lexicon, the third column shows the confusability of the lexicon, i.e. the average phone-level confusion over all words in the training data. The final column shows how long it takes to run a recognition test using the specific lexicon. It is expressed in N times real time (x RT).

Lexicon	# entries	Confusability	Timing (x RT)
1_Baseline	1198	1.5	4.5
2_PhonRules	2066	1.7	6.1
3_PhRec	20347	65.9	48.7
4_PhRec_Sel	2682	4.4	9.4
5_Dtree_Sel	4184	2.7	12.3
6_ PhRec_Sel_100	2558	2.1	8.5
7_ PhRec_100	15424	3.1	29.7

Table 2: Size of lexica, average confusability in the lexica, decoding time: N times real time (RT).

Note that confusability does not correlate that well with timing, this indicates that we may want to include other decoding influences, such as the language model in future revisions of this measure.

5.2. WER

Table 3 shows the results in terms of WER for the various lexica on the VIOS test set.

Lexicon	WER
1_Baseline	10.7
2_PhonRules	10.5
3_PhRec	10.9
4_PhRec_Sel	10.6
5_Dtree_Sel	10.0
6_ PhRec_Sel_100	10.6
7_ PhRec_100	10.1

Table 3: WER results for the different lexica.

(1) The baseline result obtained with the ICSI recognition system is an improvement compared to the results, which have been previously found in [8]. (2) Incorporating five phonological rules in the recognition process leads to a small improvement. Unlike the results obtained in [8] adding variants to the language model or retraining the acoustic models (in this case the neural net) does not lead to an additional improvement (results not shown here).

(3) Using the phone recognition output to generate a lexicon leads to deterioration in WER compared to the baseline. The deterioration is not as large as one might expect, but it should be kept in mind that the lexicon does not only contain variants from phone recognition because, like all the other lexica, it is merged with the baseline lexicon. As one would expect though, and as can be seen in Table 2, the decoding time is greatly increased. (4) Making a selection of the phone recognition data before generating the lexicon improves the WER to about the same level as the baseline result. (5) When in addition to this, d-trees are used to smooth the phone recognition prior to lexicon generation, a significant improvement compared to the baseline is found. (The result is significant at the 0.02 level using a difference of proportions significance test.)

The results of pruning the most frequently confused variants (those with a confusion count of more than 100) leads to a substantial improvement in (7) but no improvement at all in (6). It seems removing the most confusable variants from a lexicon that is already based on a pre-selection of phone recognition output does not have much effect in terms of WER, even though the average confusability in the lexicon is halved. However, when the raw phone recognition lexicon is used to calculate confusability and all variants with a confusability count higher than 100 are removed from the lexicon, the drop in average confusability is extremely large and the resulting WER is a significant improvement ($p < 0.05$) compared to the baseline result. In addition to this, the decoding time is reduced by 1.6.

It would appear; at least for the simple d-trees we are using that removing confusable variants via the confusability metric is roughly as effective as smoothing via d-trees.

5.3. Comparison between Lexica

Table 4 shows the overlap between the phonological rule lexicon and two of the data-derived lexica: phone recognition and d-trees, respectively. The number of variants generated by each of the phonological rules is shown in column 2 of Table 4. Combi indicates those variants that are the result of a combination of rules applying to a word. For the phone recognition and the d-trees lexicon, the number of variants for each of the rules was determined by comparing them to the phonological rule lexicon and counting the overlap. In columns 4 and 6, the percentage of variants in the phonological rule lexicon that is covered by the phone recognition and d-trees lexica, respectively, is shown.

	Phon.rules	Phone rec.		d-trees	
	# vars	# vars	%	# vars	%
/n/-del	283	35	12%	83	29%
/r/-del	240	33	14%	71	30%
/t/-del	61	9	15%	19	31%
Schwa-del	18	1	6%	0	0%
Schwa-ins	64	1	2%	2	3%
Combi	201	10	5%	22	11%
Total	867	89	10%	197	23%

Table 4: Number of variants present in the phonological rules lexicon, as a result of phone recognition, and after smoothing phone recognition with d-trees. Percentages indicate the proportion of variants in the phonological rule lexicon that is covered by the other two lexica.

Table 4 shows us that in total 10% of the variants present in the phonological rule lexicon are also found in the phone recognition lexicon. Using d-trees to smooth the phone recognition leads to 23% overlap between the phonological rule variants and the data-derived variants. This indicates that the d-trees are learning phonological rules. Therefore, in a further study we will investigate the effect of adding more linguistic information to the d-trees.

6. CONCLUSIONS

In this paper, we employed two different approaches to dealing with pronunciation variation. Our baseline performance is an improvement on what was found previously in [8], although, we did not find a significant improvement using the knowledge-based approach to generate new variants, as was the case in [8]. As far as the data-derived lexica are concerned, using the phone recognition output to add new variants to the baseline lexicon led to deterioration in WER. This was to be expected because of the noise that is present in the phone recognition. Removing some of the errors by pre-selecting the utterances used for generating the lexicon brings the WER back down to the level of baseline performance. Taking this lexicon and subsequently smoothing the phone recognition using simple d-trees before lexicon generation leads to a significant improvement compared to the baseline. Finally, we found that using the confusion metric to prune variants results in roughly the same improvement as using the d-tree approach.

The comparisons we made between the phonological rule lexicon and the data-derived lexica showed that some of the variation described by the five phonological rules is also found in the data-derived lexica. Using d-trees results in more overlap with the phonological rules than the phone recognition does.

Finally, we conclude that although the metric that we proposed for measuring confusability in the lexicon can be quite helpful it is definitely not perfect and in the future we want to extend it to make it a more useful tool in the process of modeling pronunciation variation.

7. REFERENCES

1. Strik, H. & Cucchiarini, C. (1999) Modeling pronunciation variation for ASR: A survey of the literature. *Speech Communication* **29**, 225-246.

2. Riley, M., Byrne, W. Finke, M., Khudanpur, S., Ljolje, A. McDonough, J., Nock, H., Sarachar, M., Wooters, C. & Zavaliagkos, G. (1999) Stochastic pronunciation modelling from hand-labelled phonetic corpora. *Speech Communication* **29**, 209-224.

3. Fosler-Lussier, J.E. (1999*) Dynamic pronunciation models for automatic speech recognition* Ph.D. thesis, University of California, Berkeley, 1999.

4. Sloboda, T. & Waibel, A. (1996) Dictionary learning for spontaneous speech recognition. *ICSLP-96*, Philadelphia, 2328-2331.

5. Torre, D., Villarrubia, L., Hernández, L. & Elvira, J.M., (1997) Automatic alternative transcription generation and vocabulary selection for flexible word recognizers, *ICASSP-97*, Munich, 1463-1466.

6. Holter, T. & Svendson, T. (1999) Maximum likelihood modeling of pronunciation variation. *Speech Communication* **29**, 177-191.

7. Strik, H., Russel, A.J.M., van den Heuvel, H. Cucchiarini, C. & Boves, L. (1997). A spoken dialog system for the Dutch public transport information service. *International Journal of Speech Technology*, Vol. 2, No. 2, 119-129.

8. Kessens, J.M., Wester, M. & Strik, H. (1999). Improving the performance of a dutch CSR by modeling within-word and cross-word pronunciation. *Speech Communication* **29**, 193-207.

9. Weka-3 Machine Learning software in Java, http://www.cs.waikato.ac.nz/ml/weka/index.html

10. Bourlard, H. & Morgan N. (1993) *Connectionist speech recognition: a hybrid approach.* Kluwer Academic Publishers.

11. Hermansky, H. (1990) Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustic Society of America* **87(4)**, 1738-1752.